exPIERience

Clients:
Crow White
Jason M. Felton
Lynne Slivovsky

Members:
Lukas Suess
email: lsuess97@gmail.com
Caleb Biggers
email: calebbiggers@gmail.com
Chris Pestano:
email: pestanochris@gmail.com
Gabriel Nguyen:
email:gschoolngu543@gmail.com

Team Email:
capstone.expierience@gmail.com

Table of Contents

1
Introduction 3
Clients/Customers 3
Stakeholders 4
Framed Insights and Opportunities 5
Project Goals and Objectives 6
- **Project Goals:** 7
- **Project Objectives:** 7

Project Outcomes and deliverables 7
Background 8
Engineering Specifications 8
Personas 10
Use Cases 10
Design Development 11
Final Design Concept 14
Prototyping 16
Final Detailed Design 21
- **High Level Scene Overview** 21
- **Pier Scene Hierarchy** 22
- **Dive HUD** 23
- **New Spawner** 24
- **Fish Animation** 25
- **Placed Fish Pathing** 26
- **Wolf Eel Animation and Implementation** 28
- **Solo Fish Movement** 28

System Integration, Testing, and Analysis 29
- **FMEA (Failure Mode Effects Analysis)** 1
- **DVP + R and Analysis** 1
- **Future Work** 1
- **System Analysis** 1

Management Plan 1
Appendices 1

Introduction

The goal of this project is to create a realistic diving simulation to allow users to see what it is like to dive beneath the Cal Poly Pier. As it is now, the simulation includes a model of the pier, a few species, and the framework for user movement. These features were created by the previous project teams. Our team will make additions to the existing project such as more creatures and life. Before any additions are added, the existing code must be optimized. This project will show users how scientific surveys can be conducted, so the ability to conduct virtual transects will be added. These additions will show users what being a diver is like and how surveys are conducted underwater.

Clients/Customers

This project is being developed for the Center for Coastal Marine Sciences (CCMS). The Center works with students and faculty to conduct marine research of the Morro Bay ecosystem and is located at the Cal Poly Pier. Our contacts from CCMS are Jason Felton, the Pier Technician and Diving Safety Officer, Dr. Crow White, an assistant professor for Cal Poly's Marine Biology Department, and Lynne Slivovsky, our Capstone professor.

The Cal Poly Pier(Figure 1) was built in 1984 by Unocal Corp. to facilitate oil transport in the US West Coast. After being donated to Cal Poly in 2001, the pier has been converted into a functioning research station with a classroom, labs, a conference room, and monitoring instruments. By the end of this project, users will be able to experience what it's like to dive beneath the pier and conduct research.

Figure 1: The Cal Poly Pier

Stakeholders

The two main uses of this project will serve to train divers by learning the pier layout before they go diving, and for educational purposes to teach people what its like to dive and conduct surveys underwater. Research institutions are one stakeholder in this project as it could be used and built upon for training in their own environment. Educational facilities can use this tool to teach students about diving. In addition to diving, the various virtual measurements taken underwater can be used to teach other mathematical concepts. In addition to institutions, this project will be used as a demo to inform the general public about the Cal Poly Pier by introducing them to marine biology and the importance marine life has. By teaching the general public the importance of marine life, in effect they may contribute to conserving marine life.

Framed Insights and Opportunities

We learned from our clients the primary use of the virtual reality (VR) exPIERience is to entertain children with the goal of giving them some enthusiasm for marine biology. However, the end goal would be to create a simulation realistic enough to give potential divers some training so that they know what to expect when underwater and train marine biologists on how to conduct surveys. This is a lofty goal and our clients know this so for the first iteration of the current team's version will be limited to a few goals given to us by our clients that are to be specified in this section.

Our first of three clients is Lynne Slivovsky. She gave us the general overview of what all three clients would like to see. Primarily we discussed the need for more marine organisms to be added to the environment which created a even more pressing issue of optimizing the rendering of the game space. The groups in the past have had trouble adding organisms due to the amount of computing power required, there were FPS(frames per second) drops noticed when the game space is too full. Additionally, Lynne expressed a need for scientific measurement tools in the simulation. These are the more broad strokes that are narrowed down when we continued the discussion with our other two clients.

In our discussions with Crow White we narrowed down the goal of scientific tools down to a transect that could be placed on fixed locations within the game world, primarily on the piling. The need for a recording method arose when discussing the transects, we settled on a button press that would increment a counter, to count the number of organisms, and record the depth at the moment of that recording. Next we decided that populating the pylons was a necessity and he would like us to add a few more species of organisms. He would also like to see some form of dynamic change of the environment with changing pollution and climate change, impacting the visibility of water and the population density of organisms. Professor White also gave us a list of a few possible organisms that we could choose from to model and add to the simulation. The need for gauges was evident when we were discussing the realism of the simulation. We decided on some form of HUD that would display information such as: depth, oxygen pressure, and a compass. The final objective that Crow said he would like to see was an updated pier model due to construction that has changed the landing platform for divers.

The next client is Jason Felton who was able to get us the plans for the new iteration of the pier in order to update the model. He described in more detail what it's like to dive and what you would potential see. The pylons were also a primary focus for him and he gave us a few species to look up and potential add to the pylons in order to populate them. We will be meeting him at the pier in order to see it in person and get he'll give us a tour pointing out what is pertinent to our project.

ExPIERinece's stakeholders and community partners are the users and other employees of the pier, disabled people, and people who live inland. ExPIERience has the potential to give people the opportunity to dive underwater when they normally could not.

Project Goals and Objectives

With our clients and stakeholders in mind we come to the goals and objectives, these summarise what we believe and what our clients have said, should be implemented overal. The goals are broad strokes that define where we end up. For example, we need to make the simulation more lifelike and add educational features. These goals can be achieved by completing the smaller and more specific objectives. In order to make the simulation more lifelike we need to add more marine life and to make it educational some scientific tools like a transect would help.

Project Goals:

1. Make the tour run better
2. Make the environment more lifelike
3. Improve user experience
4. Add educational features

Project Objectives:

1. Optimize current code to speed up performance
2. Add UI/HUD
3. Add more specific species onto the piling
4. Add scientific tools, such as a transect for the piling and benthic areas
5. Add more behaviors to animals, such as general movement and when the diver is approaching the animal

Project Outcomes and deliverables

The features of the project given to us were already enough to create a working tour, and due to the nature of the project as a whole, which can always be improved and made to be more lifelike , there is no clear point when the project is complete, other than when our clients decide enough features have been implemented.

The project outcomes of our group can be grouped into making the user experience better and adding on additional features, so by the completion of our Capstone project our group aims to deliver unto the next generation of exPIERience developers an improved version of the tour that runs faster and smoother than the version we received, with a wider variety of animals that occupy more of the environment. To provide a goal for making the game run smoother we set an initial goal of improving the average frames per second of the tour by 25%. Our clients Jason Felton and Dr. Crow White provided us a list of animals they would like to see added to the tour. We implemented the necessary pathing scripts and animal prefabs to add them successfully.

Background

We started this project with plenty of documentation about the existing game/simulator. The info has helped significantly. There are other similar diving games but none that really fully immerse you in a realistic diving simulation for a biology lesson and a diving lesson. None of our team has any experience in diving or VR games, however, most of us were at least familiar that Unity existed but have no experience with it. We all were familiar with previous coding environments and knew other object oriented languages similar to the C# that Unity uses. Some of the members of the current group have even made a game before but not in Unity and not nearly as complicated as a whole VR simulation. The biggest hurdle faced was creating 3D models of animals, no member had done any 3D modeling before which made this process slow to start.

Engineering Specifications

For this project we will be creating a virtual diving experience. The purpose of this project is for it to serve as a diving simulator for pier training, as well as a simulation of the methods marine biologists use to capture data, so users during the pier's open houses can learn about and appreciate the subject more. The goals for our group are to first optimize the simulation, then add more animals to the pier, add virtual diving and measurement tools, and to update the pier model to reflect the landing changes being constructed currently.

#	Parameter	Target	Tolerance	Risk	Compliance
1	Frame Rate	60fps	Min	M	I, T
2	Total Number of Species	7	Min	L	I
3	Piling Stratification Zones	4	Min	M	I
4	Number of Tools	2	Min	M	I
5	Accuracy of Pier Model	99%	Min	M	I

Figure 2: Engineering requirements

The engineering requirements for these goals are shown above in Figure 2. The performance requirement is to measure how optimized the simulation is. A frame rate of sixty frames per second, which describes how fast the screen updates, is considered standard in terms of a visual application like this, as it provides the user a smooth viewing experience. Frame rate is directly affected by how fast the program runs, as inefficient code and too many visual assets rendered in at once can make the program take longer to update frames, so it a good way to check for performance. We will be adding 4 additional species to the pier simulation on top of the 3 already present. Stratification zones describe how different species inhabit different areas, due to each species' ability to stay out of water, feeding habits, and defense against predators. To accurately show the diversity of the life under the pier, we will be recreating the stratification zones on the pier pylons. In order to be able to demonstrate how marine biologist conduct transects underwater we will be implementing two tools. We will first add a diving HUD to show oxygen and depth, as well as a virtual transect tool to allow the user to create transects of pylons and along the ocean. Lastly, to make sure the simulation is as close to the Cal Poly pier as possible, we will be updating the pier model with the new landing.

Personas

- A child between the ages 12 to 17 who wants to have fun by looking at species underwater and experience how it is to dive.
- An adult interested in marine biology and needs realistic and educational interaction with underwater species.
- People who live inland or those with disabilities and are incapable of participating in a real dive.

Use Cases

- The first use case to using the VR diving experience for educational purposes. This is so that divers who will be diving at the pier can get familiar with the environment before diving for real. The simulation contains research tools and is a realistic recreation of the pier for these cases.
- The second use case is as a demonstration tool for the Cal Poly Pier. This is to show the general public and get them interested in marine biology. This could also be used in classrooms to teach about marine biology.

Design Development

Below are some of the early models for fish and wildlife. Most of them are a work in progress but the finished versions are shown the next section. Figure 3 shows the early stage of the rockfish, at this point it is lacking fins and a texture. This was the first model anybody had made. The other figures in this section are showing the leopard shark, wolf eel, starfish and some sleeves for the pilings. All models are a work in progress.

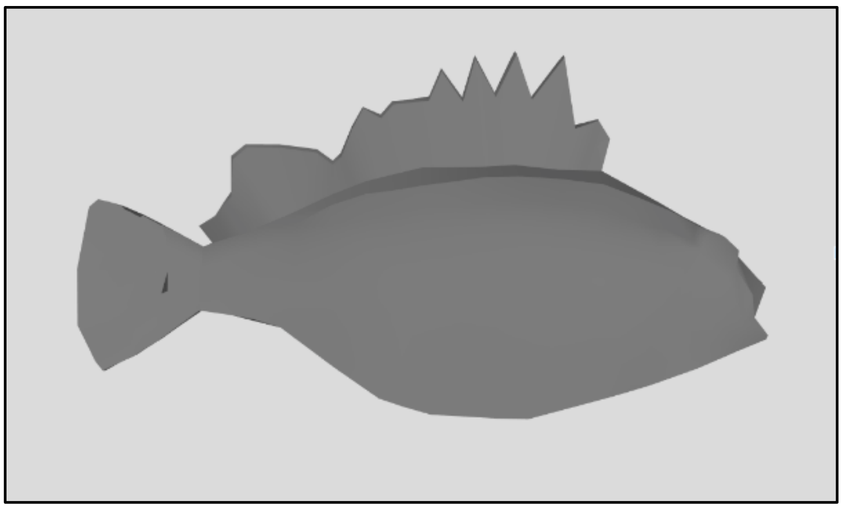


Figure 3: Untextured Early Model of Gopher Rockfish

Figure 4: Uncolored Finished Model of Leopard Shark

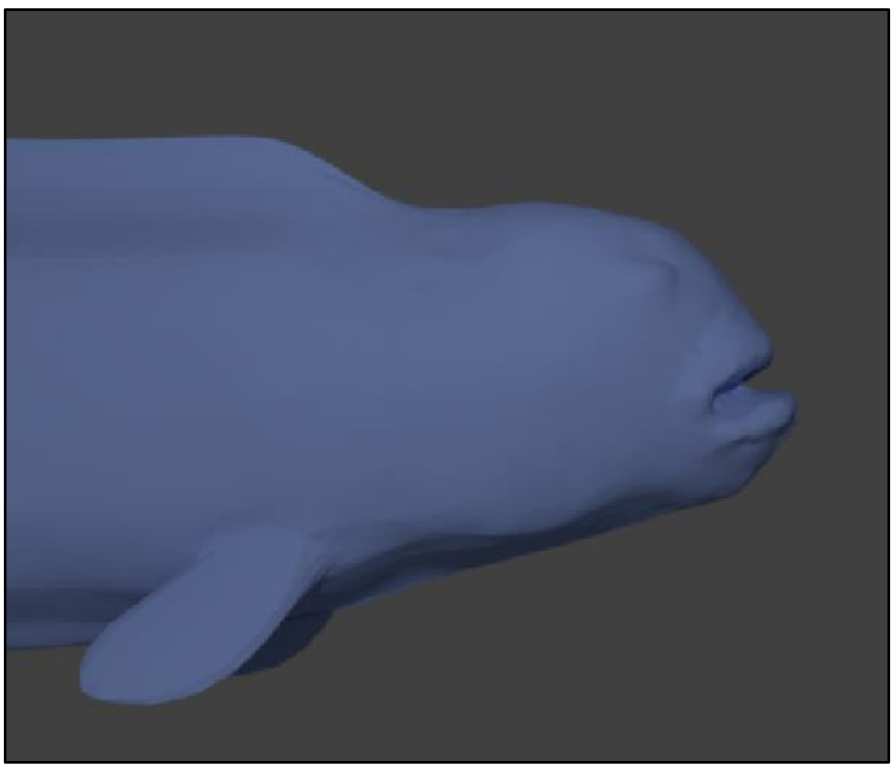


Figure 5: Untextured Early Model of Wolf Eel

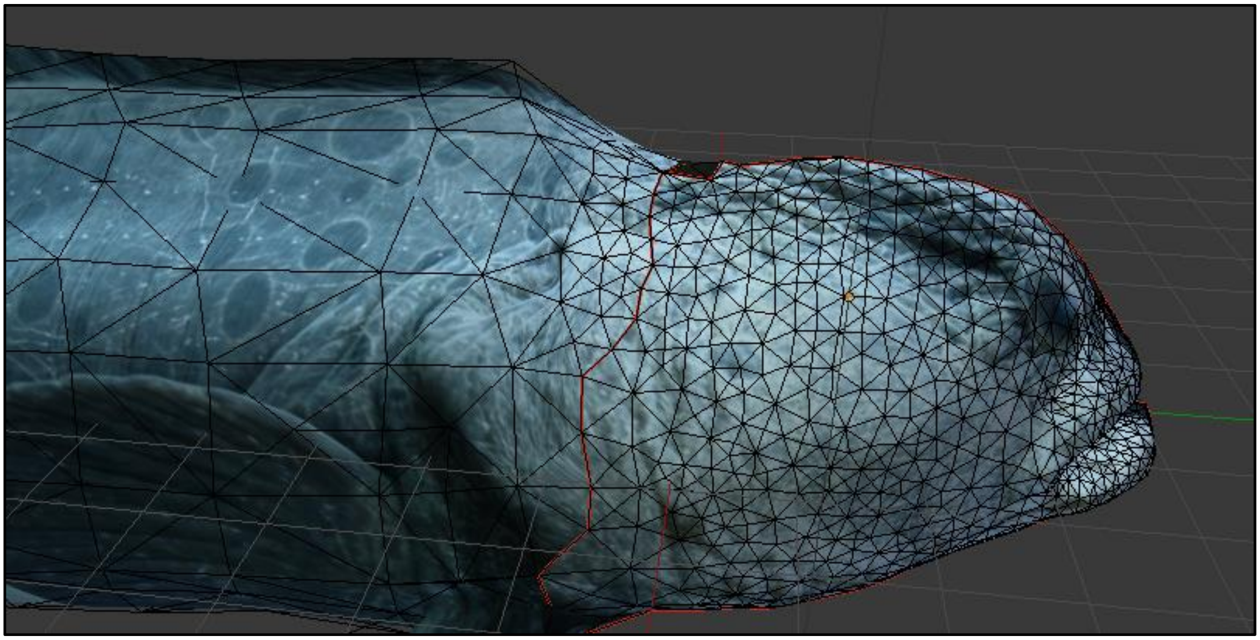


Figure 6: Unfinished Texturing of Wolf Eel

Figure 7: Barnacle Cluster

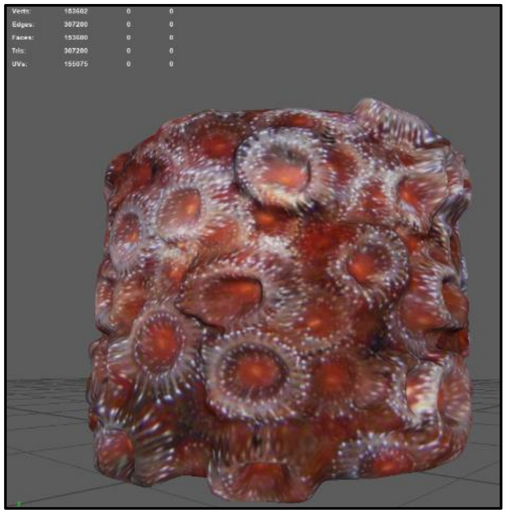


Figure 8: Strawberry Anemone Clone Cluster

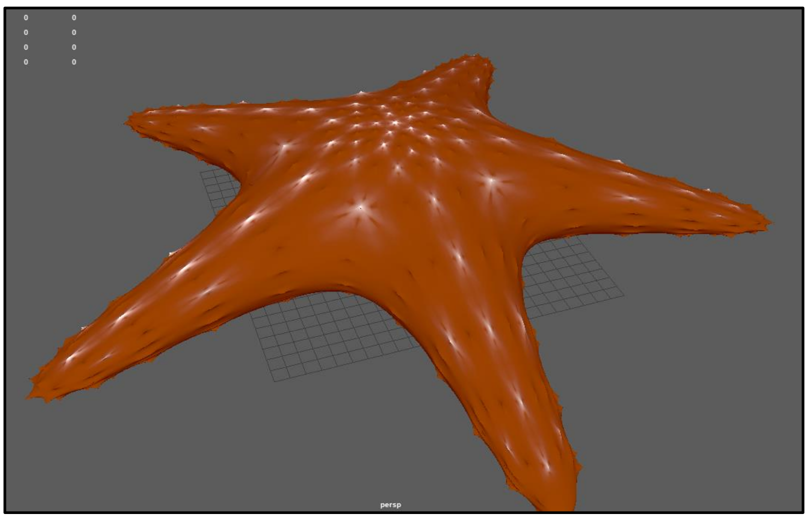


Figure 9: Giant Sea Star

Final Design Concept

The Figures 10 and 11 below show the end result of the models that are fully complete. The rockfish is now textured and modeled completely and the leopard shark’s texture is altered to make it more brown after some errors were pointed out by our clients.

Figure 10: Finished Model of Gopher Rockfish

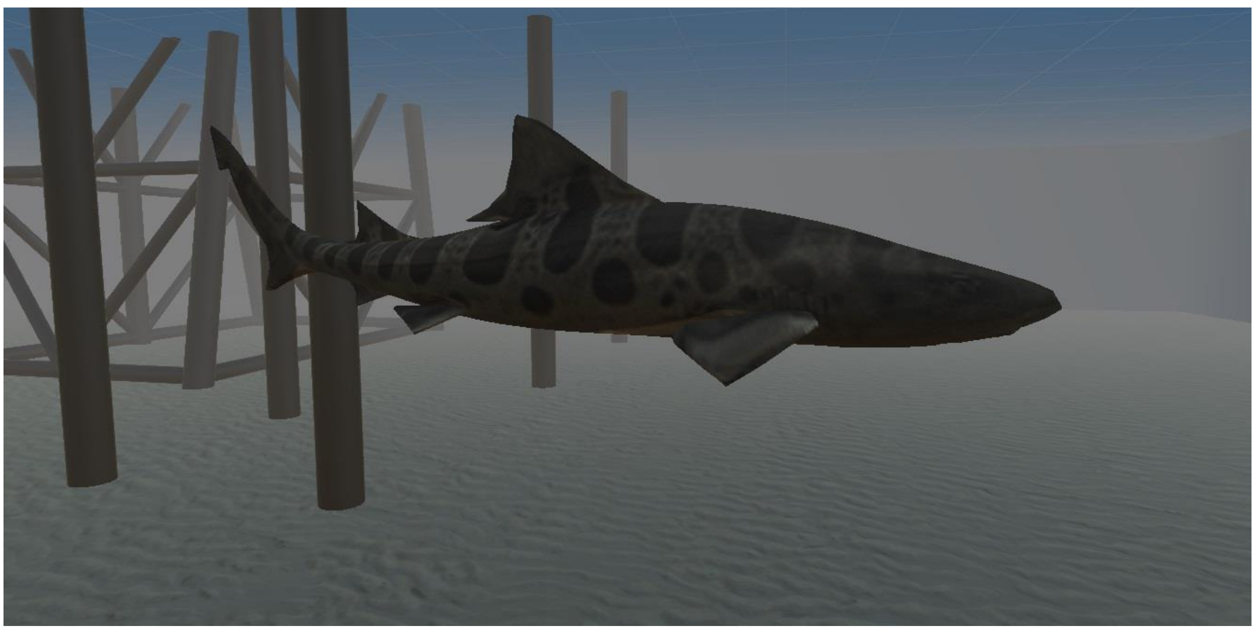
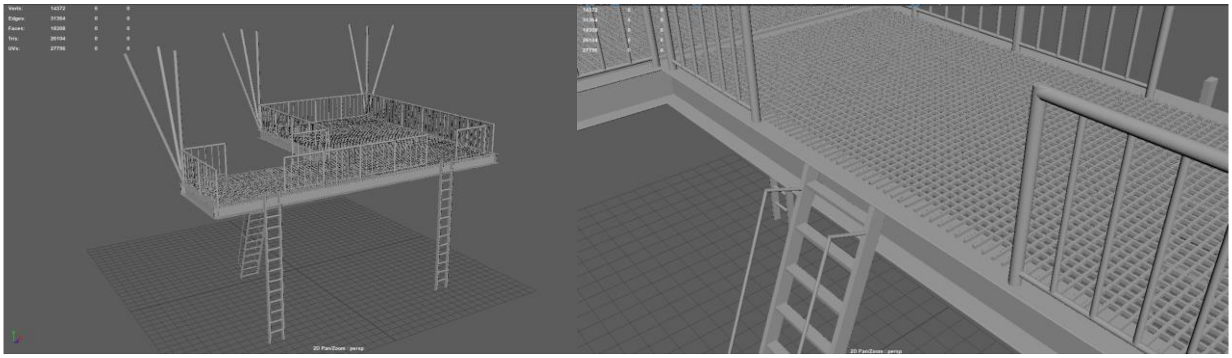


Figure 11: Finished Model of Leopard Shark

Figure 12: Finished Model of Cabezon

Prototyping

This sprint a major model we chose to implement was the planned renovation for the pier's landing, where the player spawns. The model is built to scale based on the schematics provided by Jason Felton and the major parts of the landing that are key to its appearance. The guardrails, hangers, ladders, and floor grating were all included in the model. However, to save performance some details were excluded to keep the polygon count at reasonable levels. The fenders and mounts for the hangers, piling, and beams all had complex shapes, so they were left out. Additionally, bore holes and small outcropping to major parts that wouldn't be noticed and offered little to the overall appearance of the pier were also excluded. The model doesn't have texture, but since most of the pier is comprised of steel and aluminum most of the pier will just be a shade darker or light, so the general color is fairly close to the final project. The model currently has issues getting imported into Unity from Maya, as the scaling changes to either about 1/27th or 30 times the size it should be, depending on if the .fbx file or .mb versions of the model are used.



Figures 13 and 14: Unskinned models of the new landing

There were two steps needed in order to start animating the Gopher Rockfish, rigging and skinning the model. The rigging of the Gopher Rockfish can be seen in Figure 15, the model needs a bone structure in order to easily move parts of the fish such as the fins or the tail. However, rigging doesn’t work by itself and needs to be skinned as well, weights need to be added onto each joint. For example, when moving one of the fishes fin, without adding or replacing weights on the fin, then not all of the fin moves along with the fin’s joint. An example of skinning can be seen in Figure 16 below displays a color ramping of the skinning of the tail’s joints. The more red an area of the Gopher Rockfish is, the more weight it has on the joints, if the color is black/blue then there is no weight on the joints. The tail’s joint will have no effect on the upper half of the Gopher Rockfish’s body since there is no weight added.

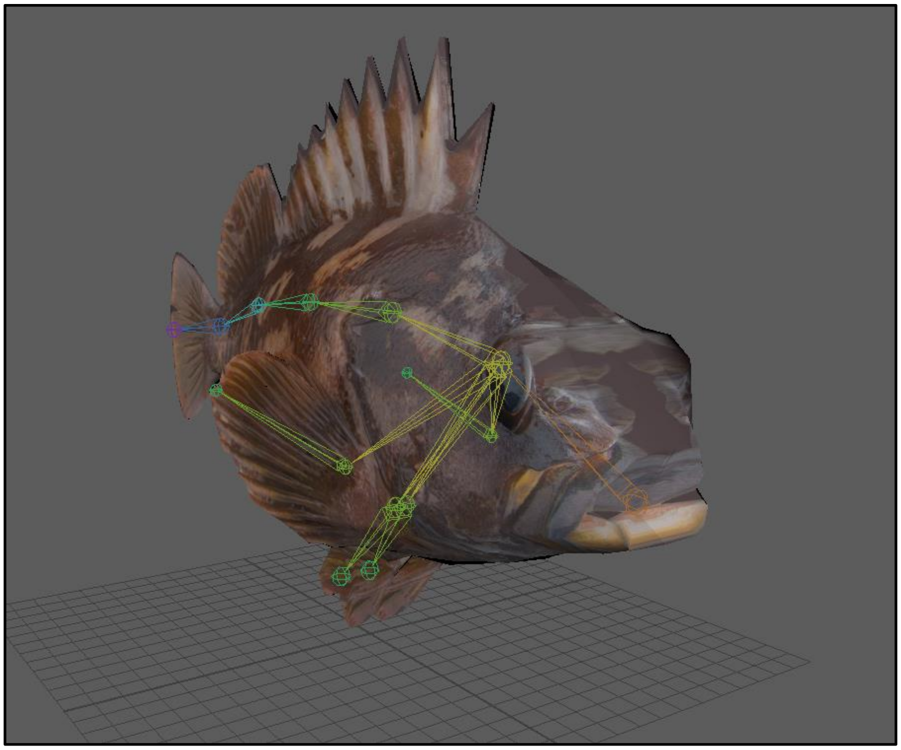


Figure 15: Rigged model of the Gopher Rockfish

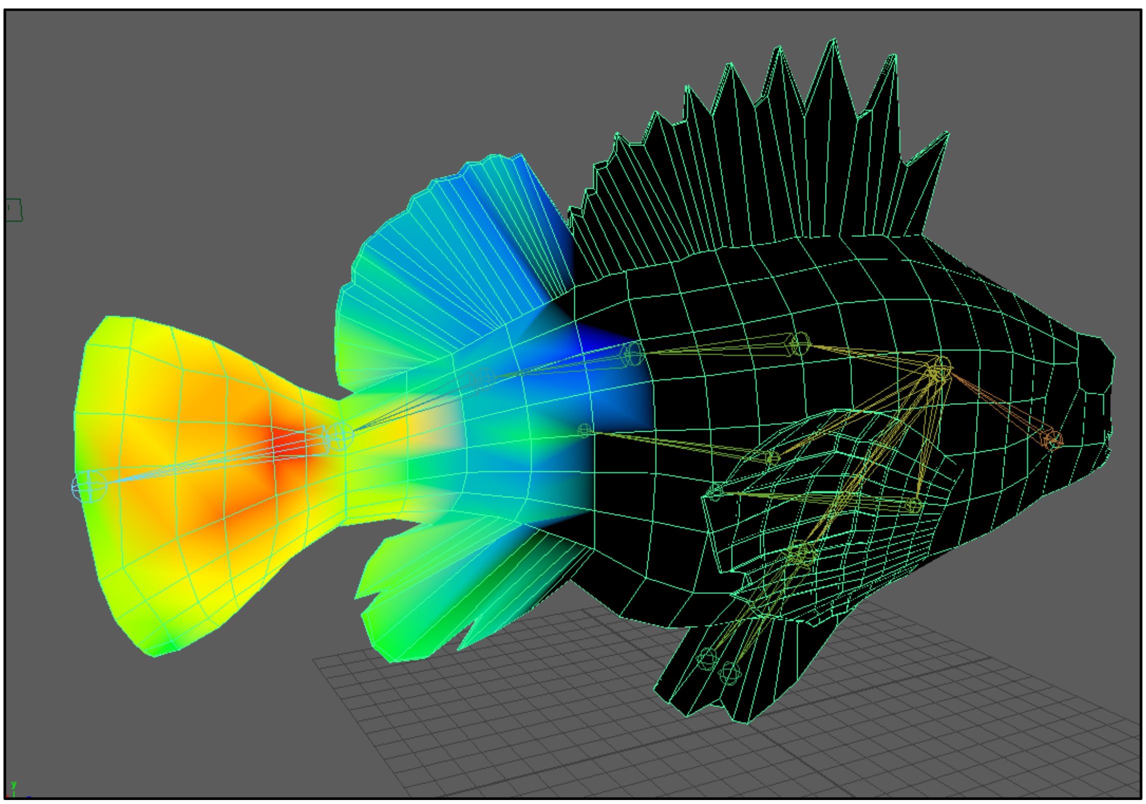


Figure 16: Weight Map of Skinned Gopher Rockfish's Tail

In addition to the new landing and fish animations, the HUD is something we worked on for our prototyping. The HUD is modeled after a diving gauge, but is displayed in front of the user in the simulation in 2D. The HUD contains three portions: a depth gauge, a gauge for air remaining, and a compass. The depth gauge is controlled by the depth of the player's head in the simulation. The air gauge is set to a default value that can be changed and it goes down by a set rate that can also be changed. This air gauge is intended to be used as a method of keeping the user in the simulation for a set amount of time. Lastly, the compass is controlled by the direction that the players head is looking. Figure 17 below shows the HUD that is rendered in the simulation.

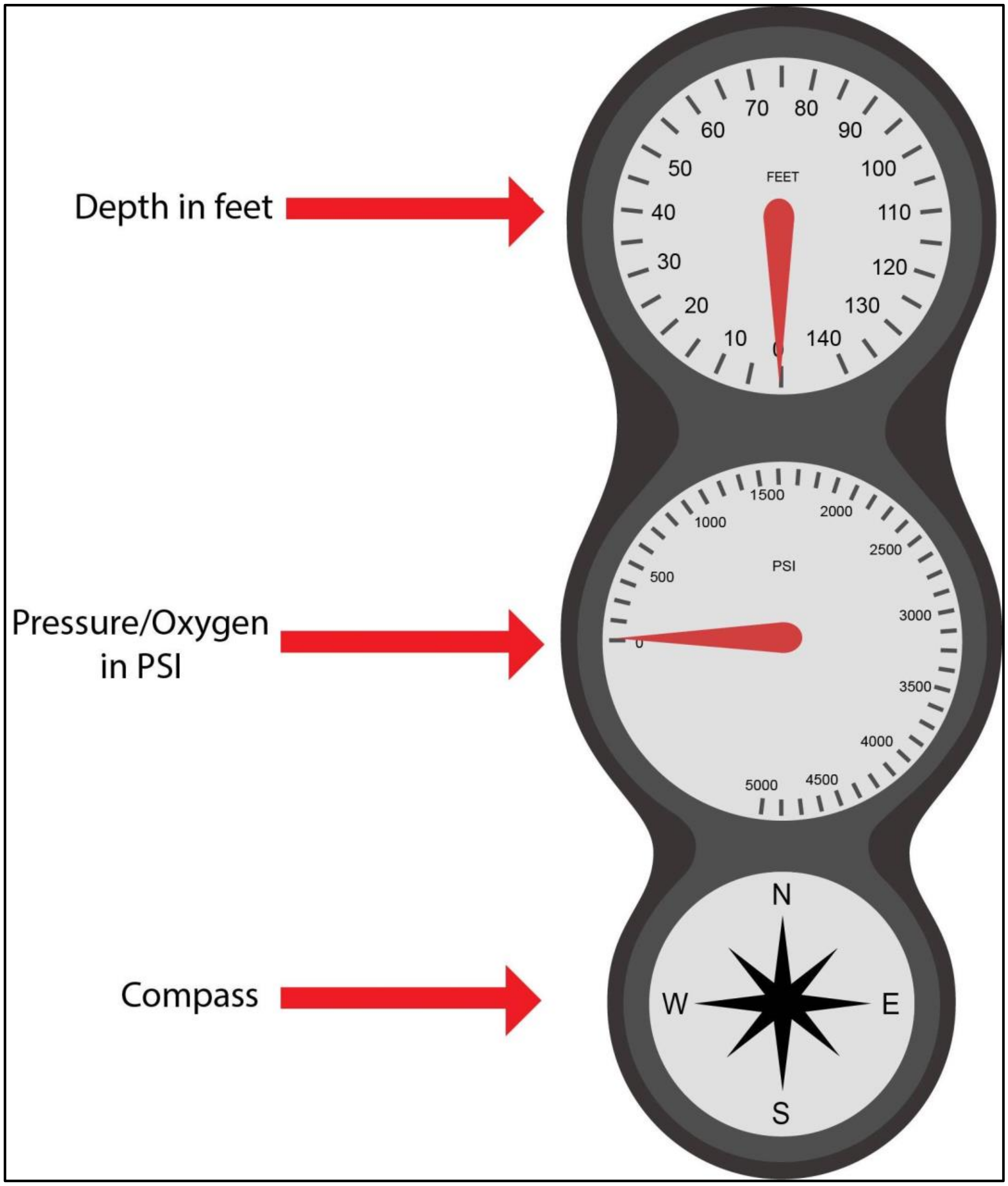


Figure 17: Diving gauge HUD

A new creature was added to the simulation in the form of the wolf eel. The texture had to be created in photoshop by merging multiple images of wolf eels and constructing the correct texture. The wolf eels come in pairs that are attached to a rock. The wolf eel were prototyped in a separate scene, shown below, where their scripts and appearance were perfected. The wolf eels detect the player and hide in their rock if the player is too close. Figure 18 below shows the wolf eel combo that is rendered in the simulation.

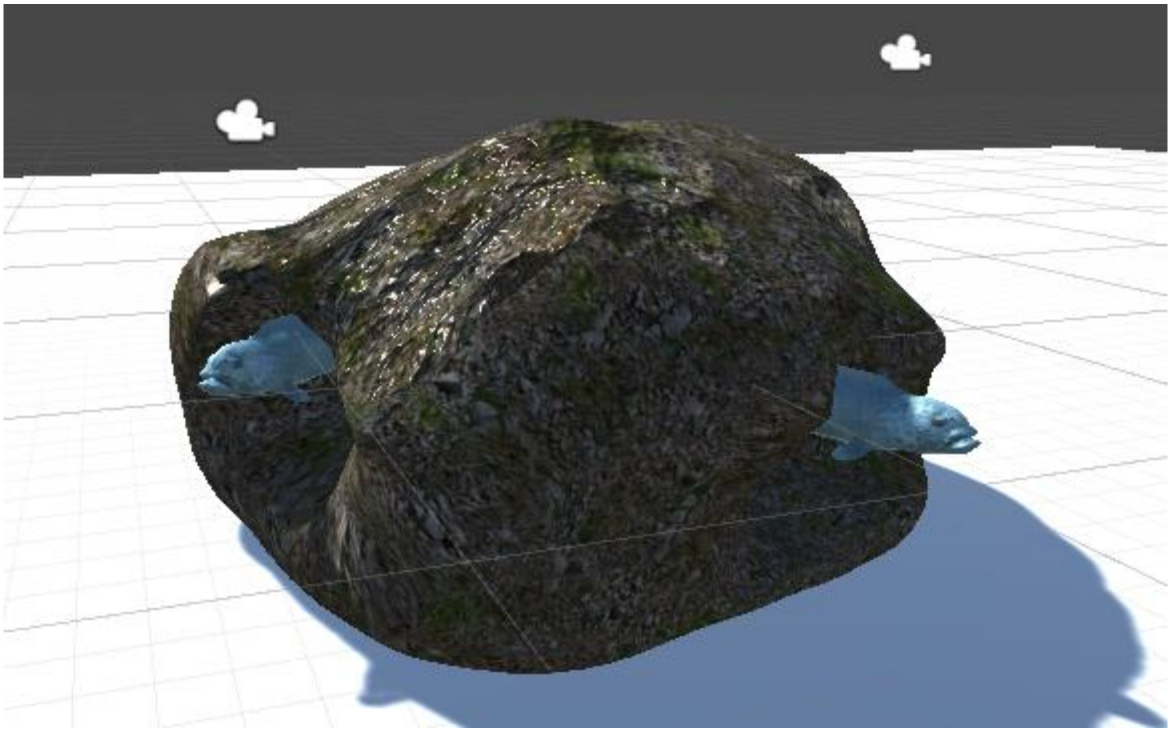


Figure 18: Wolf eel combo

Final Detailed Design

High Level Scene Overview

The project is organized into "scenes". Each scene is a kind of like a level or area that the user is in. The project has four scenes: main menu, 360 video, controls, and pier. Figure 18 below shows the organization of the scenes. The user starts at the main menu and can choose to navigate to either the 360 video or the pier diving simulation. When the 360 video is selected the user is in the middle of the 360 video while it plays. For the diving simulation, the user starts in the controls scene and can swap back and forth to the pier. The Unity engine handles loading each scene. The pier scene is the main scene in which we are editing the project.

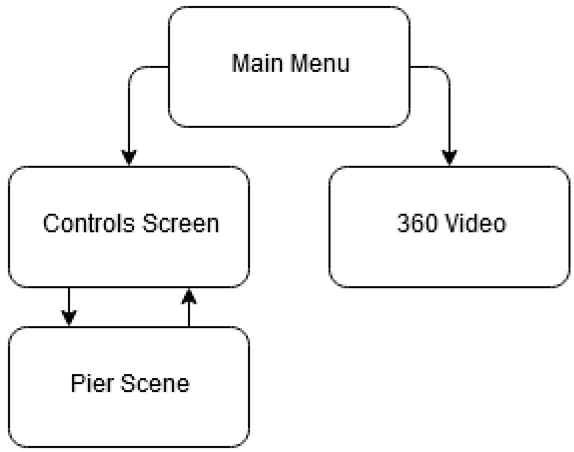


Figure 19: Scene hierarchy

Pier Scene Hierarchy

The pier scene consists of a list of game objects. Each part of the scene has its own containing object. Figure 19 below shows the hierarchy. The pier consists of individual models like the pylons, deck, and the new landing. The old fish spawner is contained in a gameobject called “fish parent”. This object starts empty and on startup spawns all the fish as children. The geoduck spawner works in the same way. The player gameobject contains the users FOV and is controlled by the player. The HUD canvas is a canvas. A canvas is a 2D plane that we can add sprites to. The HUD canvas is positioned in front of the users FOV to emulate a diving HUD. Our new fish are split into spawned fish, similar to the old fish, and placed creatures.

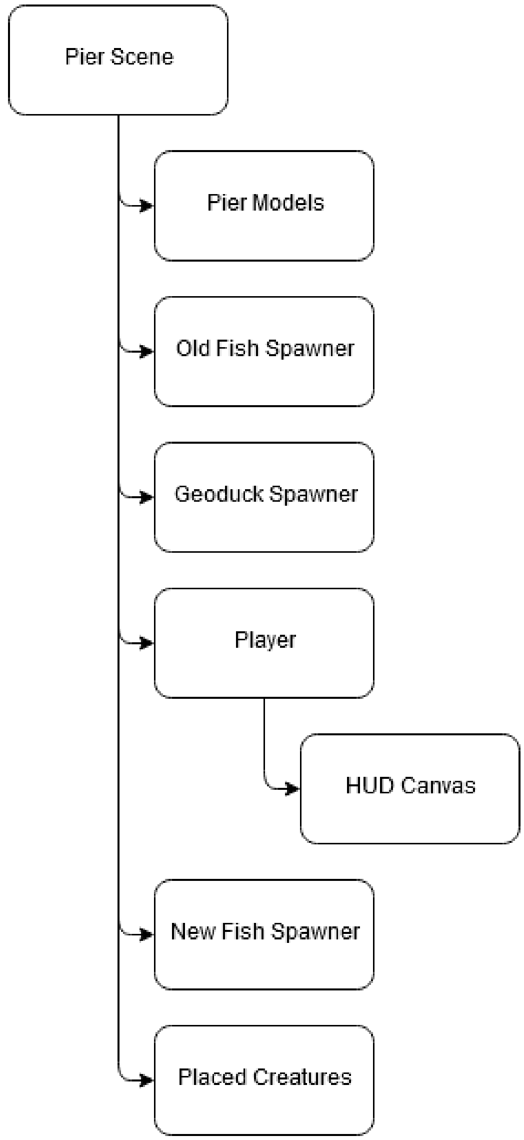


Figure 20: Pier scene hierarchy

Dive HUD

Each part of the gauge is its own sprite and is controlled by a script attached to the main canvas. Figure 20 shows this organization. The gauge background is static and does not change. The needles are controlled by the script. The depth gauge is connected to the game world depth of the user’s head. The air pressure remaining is set in the HUD controller script.

The compass face is controlled by the direction that the user's head is facing. The digital readouts are just text boxes with a digital font reading the values the needles are using.

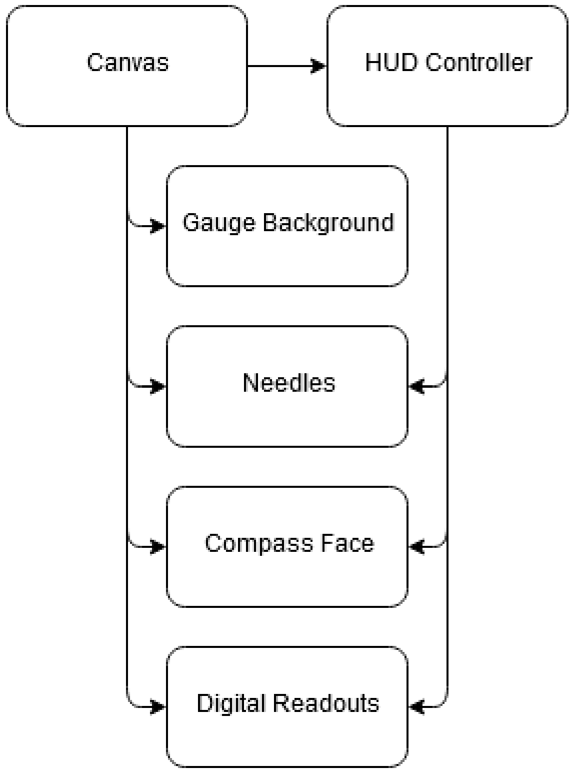


Figure 20: HUD canvas

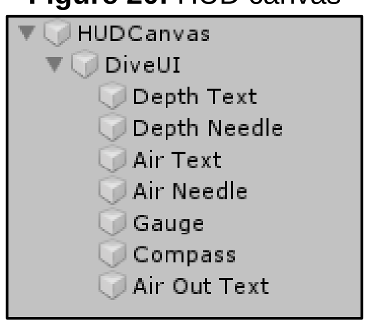


Figure 21: In Unity view

New Spawner

A new, universal spawning script was created for any fish or other future object that is intended to be spawned randomly in the scene. An instance of this spawning script can be placed within the scene, with its location acting as the origin point for its spawning zone. The script has global variables that could be modified in the Unity editor, getting rid of the need for future groups to read and edit the actual script file to suit their needs. As seen in Figure 22, the global variables are numbers of objects to be spawned, the depth (vertical) range of the spawning zone, the radius (horizontal) range of the spawning zone, minimum spacing between the spawned object and other objects already in the scene, and the object that will be spawned by the spawner. The integer values represent distance units in Unity. This spawner adds some modularity to the code, allowing us and future groups to add in new fish models and modify their spawning characteristics easier and quicker, as they can all use the same code and can change important values in Unity without opening files.

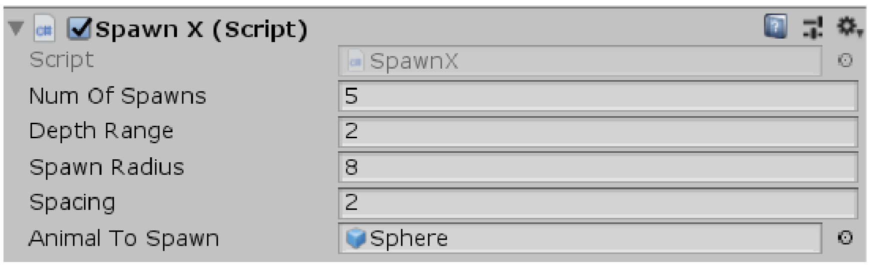


Figure 22: Spawning script global variables as seen within Unity View

Fish Animation

A model needs several different components in order to be animated. Each model needs its own bone structure, such as joints for its fins, tails, head, and even gills. The joints provide pivot points when moving each part of the fish's body. Skinning adds weighting to each joint so the joint doesn't influence other parts of the body, the tail moving shouldn't influence the head's movement. After rigging and skinning, then each joint of the fish can be given keyframes that hold the position of each joint in each frame of time.

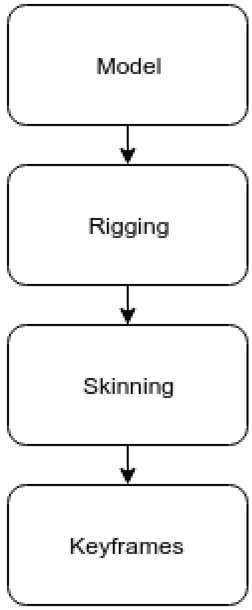


Figure 23: Fish Animation Hierarchy

Placed Fish Pathing

Our simulation has two kinds of pathing, dynamic and static. The dynamic pathing is random and was already implemented. The static pathing script is a set path that is paired with an game model to control its movement. The new pathing script is meant to be reused many times. The script is used constructed by three parts, the PathHolder object is just an empty object that will have the script to create the path attached. The Nodes are empty gameObjects that represent points. Nodes are children of the PathHolder object. A path is formed from one node to the next. Then the last part is the object to follow the path, simply attach the MoveOnPath script that moves the object from one node to the next then repeats the loop.

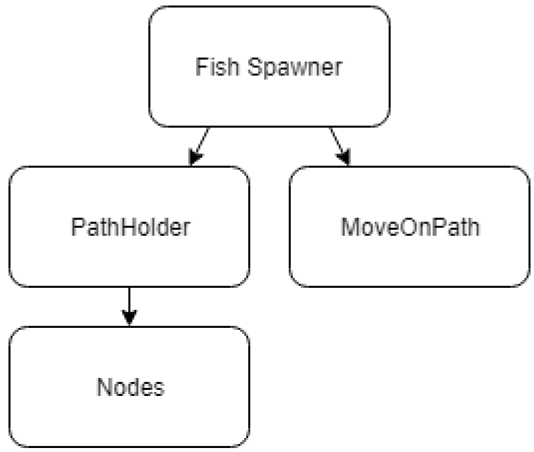


Figure 24: Placed Fish Pathing Hierarchy

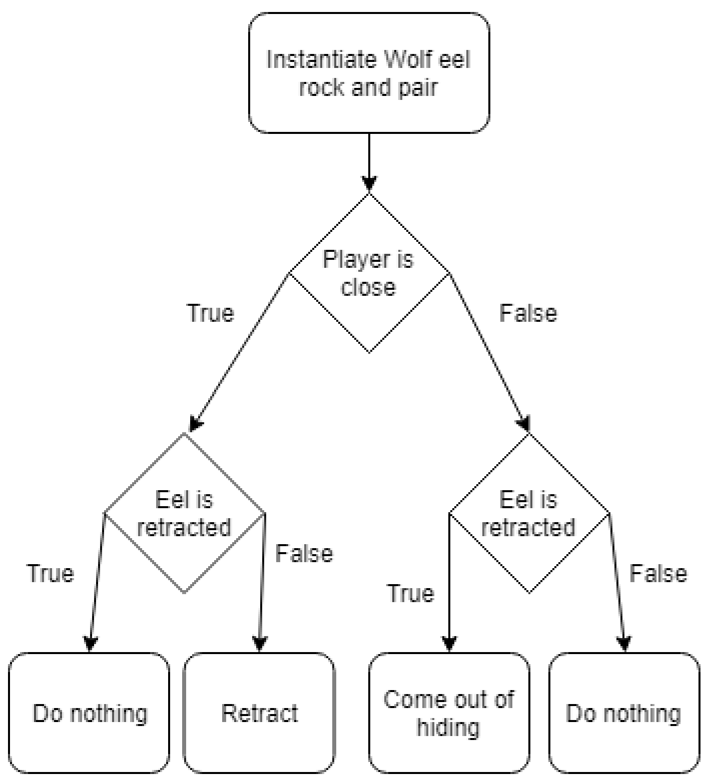


Figure 25: Placed Fish Pathing Hierarchy

Wolf Eel Animation and Implementation

Our simulation now has three sets of the wolf eel pair present in the main scene. The wolf eels are attached to a rock that was custom made with holes for the eel to retreat into. The eels have a script that when a player object is within a certain radius, the eels jump back into their holes and don't come back out until the player has left the area. The wolf eels come in pairs that are attached to the same rock and act as a prefab in Unity.

Solo Fish Movement

The two scripts responsible for controlling movement for schools of fish were combined and modified to work with the individual fish that were randomly generated by the spawner objects. The path finding for these fish work differently as rather than changing trajectory after reaching a goal position they change their pathing after a random amount of time. This is to fix an issue where some of the models were imported with the front of the model was accidentally assigned to the tail of the fish, so the fish would swim backwards with the normal code. Having the fish movement update by time rather than reaching goals positions allows for an easy fix for this, as for the fish models that are imported backwards the transform function that control movement just needs to be multiplied by -1 for the appropriate axis. The cabezon is the first fish to utilize this method in scene.

Ladder Climbing

Some of the feedback we were given was that the user cannot climb back up the ladder after dropping from the landing into the water. In order to solve this problem, we added functionality to the ladders oh the landing. This is accomplished with 2 bounding boxes attached to each ladder. Figure 26 below shows the bounding box placement. One box is attached to the base to detect when a player is trying to climb the ladder, and the second covers the whole ladder to detect when the player has left the ladder. The climbing is accomplished by disabling the movement script attached to the player when the enter the bottom bounding box. While on the ladder, the rise and sink buttons allow the player to climb the ladder.

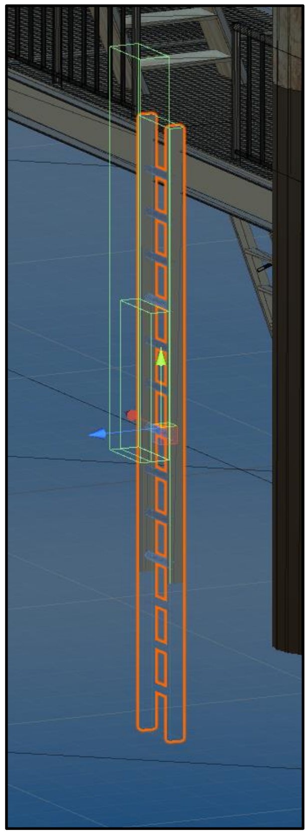


Figure 26: Ladder bounding boxes

System Integration, Testing, and Analysis

FMEA (Failure Mode Effects Analysis)

FAILURE MODE AND EFFECTS ANALYSIS

Item:	exPIERience	Responsibility:	-	FMEA number:	-
Model:	Current	Prepared by:	-	Page :	1 of 1
Core Team:	Lukas Suess, Caleb Biggers, Chris Pestano, Gabriel Nguyen			FMEA Date (Orig):	1/31/2019 Rev: 1

Process Function	Potential Failure Mode	Potential Effect(s) of Failure	Sev	Class	Potential Cause(s)/ Mechanism(s) of Failure	Occur	Current Process Controls	Detec	RPN	Recommended Action(s)	Responsibility and Target Completion Date	Action Results: Actions Taken	Sev	Occ	Det	RPN
Oculus	Video doesn't work	User cannot see	8		Incorrect cable set up	5	Operator training and instructions	1		Fix the wiring						0
			8		Hardware broken: screen, cable, connector	2	Possible software warnings or errors	4		Submit oculus for maintenance						0
	Audio doesn't work	User cannot hear	4		Hardware broken: headset, cable	2	Possible software warnings or errors	4								
	Unsecure headset	User cannot wear headset	7		Broken or loose straps		N/A			Get replacement parts						
			7		Incompatible headsize		N/A			No solution						
Computer	Computer freezes/shuts down	Game cannot run	8		CPU/GPU overheats	3		2		Replacement parts is needed, optimize code						
			8		Low specs/ outdated hardware	6		3		Need better computer						0
Controller	Inputs not registering	User cannot control diver	7		Controller not plugg in	3	None	1		Plug in						
			7		Controller broken	1		3		Get a new controller						0
			7		Out of battery	7		2		Charge batteries						0
Executable	Audio doesn't work	User cannot hear	4		Program muted/computer muted	4		1		Raise volume						0
			4		Sofware Issues	3		6		Debug code						0
	Game freezes	User cannot play game	8		Out of memory/core dump	2		1		Optimize code						0
	Low FPS	User experience lowered	5		Too many active assets/ poorly optimized code	4		1								
		Possible motion sickness	9		Too many active assets/ poorly optimized code	4		1								

DVP + R and Analysis

CPE 450 DESIGN VERIFICATION PLAN AND REPORT													
Report Date			Sponsor	Lynne Slivovsky, Crow White, Jason Felton				Component/Assembly			REPORTING ENGINEER:		
TEST PLAN										TEST REPORT			
Item No	Specification	Test Description	Acceptance Criteria	Test Responsibility	Test Stage	SAMPLES		TIMING		TEST RESULTS			NOTES
						Quantity	Type	Start date	Finish date	Test Result	Quantity Pass	Quantity Fail	
1	FPS	View unity FPS count while in game	Above 24 FPS	Caleb		5	Test	2/20/2019	2/20/2019	Pass	5	0	Using unity measured framerates. Averaged 180 fps on a GTX 1080 TI graphics card. Nothing else running.
2	Spawning	Check to make sure animals spawn correctly	Animals spawn within acceptible bounds and not in objects	Chris		1	Insp.	2/20/2019	2/20/2019	Pass	1	0	Verified all instantiated objects did not spawn inside other objects
3	Fish Models	Model shape and skin texture	Fish are recognizable	Gabby and Lukas		3	Insp.	2/20/2019	2/20/2019	Pass	3	0	Observed models details.
4	Fish animation	Fish move similar to their real life counterparts	Fish behave as expected	Gabby and Lukas		3	Insp.	2/20/2019	2/20/2019	Pass	3	0	Compared against fish's real life movement.
5	Fish behavior	Fish movement and animations work in game	Fish look as they do in Maya and move on their planned paths	Lukas		1	Insp.	2/24/2019	2/24/19	Pass	1	0	Used inspection to ensure proper movement.
6	Gauge	Depth, air, and compass gauge	Gauge can be read comfortably	Caleb and Lukas		1	Insp.	2/20/2019	2/20/2019	Pass	1	0	Using Oculus. Tested if the gauge is overall legible.
7	New landing	Old pier landing model replaced with landing currently under construction	Landing is put into the scene complete and at the correct scale	Chris and Caleb		1	Insp.	2/15/2019	2/15/2019	Pass	1	0	Model needed to be rescaled due to discrepency between units of measurements in Maya and Unity

Future Work

ExPIERience has two varying audiences, one group people are those just generally looking for entertainment and another group is those interested in marine biology. The current environment is sparse of fish to represent the actual Cal Poly Pier environment. Another feature could be added where there is another version purely for entertainment purposes, which would allow for a lot more fish to be added into the environment.

System Analysis

Management Plan

We are communicating on who is working on the pier structure, different species, and optimization. Caleb is our primary liaison, so he ensures we're continuously updating the clients and getting the necessary information needed in order to work on the project. We make sure to send the clients images of our work in order to make sure the work is accurate in representing the species in the pier environment.

Appendices

Estimated Prototyping budget: $0

Predicted expenses: $0

We don't have anything hardware wise since our whole project is software based. Additionally, Unity and Blender are free so we don't require any software licenses.